



SOFTWARE QUALITY ENGINEERING

Assignment 02

Submitted by: Muhammad Shamoil
Section: BSSE - V - A
Submitted to: Sir Naseer Jan
Date of Submission: October 25, 2025

Contents

<i>Effect of Software Process Model on Quality and Defect Density</i>	1
<i>Classification of Projects into Bespoke, Market-Driven, and Technology-Push</i>	2
3. Defect Handling and Classification of Projects	3
4. Quality Interpretation Across Different System Types	4
Conclusion	4
References	4

Effect of Software Process Model on Quality and Defect Density

The selection of a software process model plays a vital role in determining the overall quality of software and its defect density. A well-structured process ensures that development progresses in a systematic manner, reducing the chances of errors being carried forward to later stages.

For instance, the Waterfall model emphasizes phase completion and documentation, which helps in maintaining traceability but may struggle with changing requirements. On the other hand, Agile models such as Scrum and Extreme Programming (XP) are iterative and encourage continuous feedback, leading to earlier defect detection and lower defect density.

The Spiral and V-Model combine structured phases with verification and risk management, which enhances both quality and maintainability. In general, the more iterative and feedback-driven a model is, the higher the software quality and the lower the defect density, since testing and improvement are part of every iteration.

Classification of Projects into Bespoke, Market-Driven, and Technology-Push

Project Type	Description	Category
Replacing an existing system	Rebuilding or modernizing a legacy or outdated system to meet current business goals and requirements.	Bespoke
Enhancing an existing system	Improving the current software to increase performance, quality, or user satisfaction.	Bespoke
Geographically distributed stakeholders	Projects involving global teams and clients, focusing on large-scale collaboration and shared requirements.	Market-driven
Embedded system	Specialized software designed for hardware-specific tasks, often operating under real-time constraints.	Technology-push

Bespoke projects are custom-built for specific organizations. Market-driven projects are influenced by end-user demand and trends, while technology-push projects emerge due to advancements in technology rather than customer requests.

Defect Handling and Classification of Projects

Project Type	Defect Handling Approach	Category
Replacing an existing system	Defects are managed through regression and acceptance testing to ensure compatibility with existing workflows.	Bespoke
Enhancing an existing system	Each enhancement is tested incrementally to prevent introducing new bugs and maintain stability.	Bespoke
Geographically distributed stakeholders	Teams use issue-tracking tools, continuous integration, and frequent builds to detect and fix defects collaboratively.	Market-driven (MDRE)
Embedded system	Defects are addressed through real-time debugging, simulation, and strict verification due to critical timing constraints.	Technology-driven

In bespoke projects, defect handling involves close interaction with clients to ensure the system meets their expectations. In market-driven projects, issue tracking and user feedback loops are crucial to manage a large number of defect reports. In technology-driven projects, testing is strict and focused on reliability and performance rather than user feedback.

Quality Interpretation Across Different System Types

System Type	How Quality Is Interpreted
Embedded / Real-Time Systems	Quality is based on timing accuracy, reliability, and low resource consumption. The system must meet strict deadlines.
Safety-Critical Systems	Quality means zero tolerance for failure. Testing, validation, and reliability are the top priorities (e.g., medical or aviation systems).
Enterprise Systems	Quality focuses on scalability, maintainability, and data security to support complex business processes.
Web Systems	Quality is judged by usability, performance, security, and user experience. Frequent updates are part of maintaining quality.
Cloud Systems	Quality means availability, elasticity, and fault tolerance while ensuring data integrity and user privacy.

Conclusion

The software process model chosen sets the foundation for how well quality is maintained throughout the project. Iterative and customer-focused approaches like Agile tend to produce better results in terms of quality and defect density. Different project categories—bespoke, market-driven, and technology-push—require distinct approaches to defect handling depending on their purpose and stakeholders. Similarly, the interpretation of software quality differs across system types, focusing on reliability for embedded systems, scalability for enterprise systems, and user satisfaction for web or cloud systems.

References

1. TutorialsPoint. (n.d.). *Software Engineering — Process Models*
https://www.tutorialspoint.com/software_engineering/software_process_models.htm
2. GeeksforGeeks. (n.d.). *Software Quality — Factors, Metrics and Models*. Retrieved from:
<https://www.geeksforgeeks.org/software-quality/>
3. Perplexity and ChatGPT for Information Understanding and Summarizing.